WRDC-TR-90-8007
Volume V
Part 42

AD-A250 474

INTEGRATED INFORMATION SUPPORT SYSTEM (IISS)
Volume V - Common Data Model Subsystem
Part 42 - CDM Impact Analysis Build Instructions User's Manual

M. Apicella, S. Singh

Control Data Corporation
Integration Technology Services
2970 Presidential Drive
Fairborn, OH 45324-6209

DTIC
ELECTE
MAY 19 1992
S D

September 1990

Final Report for Period 1 April 1987 - 31 December 1990

92-13196

Approved for Public Release; Distribution is Unlimited

MANUFACTURING TECHNOLOGY DIRECTORATE
WRIGHT RESEARCH AND DEVELOPMENT CENTER
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433-6533

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever, regardless whether or not the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data. It should not, therefore, be construed or implied by any person, persons, or organization that the Government is licensing or conveying any rights or permission to manufacture, use, or market any patented invention that may in any way be related thereto.

This technical report has been reviewed and is approved for publication.

This report is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations

25 July 91

DAVID L. JUDSON, Project Manager
WRDC/MTI
Wright-Patterson AFB, OH 45433-6533

DATE

FOR THE COMMANDER:

25 July 91

BRUCE A. RASMUSSEN, Chief
WRDC/MTI
Wright-Patterson AFB, OH 45433-6533

DATE

If your address has changed, if you wish to be removed form our mailing list, or if the addressee is no longer employed by your organization please notify WRDC/MTI, Wright-Patterson Air Force Base, OH 45433-6533 to help us maintain a current mailing list.

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION Unclassified	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for Public Release; Distribution is Unlimited.
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) UM 620341421	5. MONITORING ORGANIZATION REPORT NUMBER(S) WRDC-TR- 90-8007 Vol. V, Part 42
6a. NAME OF PERFORMING ORGANIZATION Control Data Corporation; Integration Technology Services 6b. OFFICE SYMBOL *(if applicable)*	7a. NAME OF MONITORING ORGANIZATION WRDC/MTI
6c. ADDRESS *(City,State, and ZIP Code)* 2970 Presidential Drive Fairborn, OH 45324-6209	7b. ADDRESS *(City, State, and ZIP Code)* WPAFB, OH 45433-6533
8a. NAME OF FUNDING/SPONSORING ORGANIZATION Wright Research and Development Center, Air Force Systems Command, USAF 8b. OFFICE SYMBOL *(if applicable)* WRDC/MTI	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUM F33600-87-C-0464
8c. ADDRESS *(City, State, and ZIP Code)* Wright-Patterson AFB, Ohio 45433-6533	10. SOURCE OF FUNDING NOS.
11. TITLE *(Include Security Classification)* See block 19	PROGRAM ELEMENT NO. 78011F / PROJECT NO. 595600 / TASK NO. F95600 / WORK UNIT NO. 20950607

12. PERSONAL AUTHOR(S)
Control Data Corporation: Apicella, M. L., Singh, S.

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT *(Yr.,Mo.,Day)*	15. PAGE COUNT
Final Report	4/1/87-12/31/90	1990 September 30	17

16. SUPPLEMENTARY NOTATION

WRDC/MTI Project Priority 6203

17. COSATI CODES			18. SUBJECT TERMS *(Continue on reverse if necessary and identify block no.)*
FIELD	GROUP	SUB GR.	
1308	0905		

19 ABSTRACT *(Continue on reverse if necessary and identify block number)*

This document is to be used by the Common Data Model Administrator (CDMA) to determine the impact a software change might have upon other software modules within the CDM subsystem. The CDM Impact Analysis Utility is used to identify and report which software modules and external schemas are affected by a change to the CDM.

BLOCK 11:

INTEGRATED INFORMATION SUPPORT SYSTEM
Vol V - Common Data Model Subsystem

Part 42 - CDM Impact Analysis Build Instructions User's Manual

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT	21. ABSTRACT SECURITY CLASSIFICATION	
UNCLASSIFIED/UNLIMITED x SAME AS RPT. DTIC USERS	Unclassified	
22a. NAME OF RESPONSIBLE INDIVIDUAL David L. Judson	22b. TELEPHONE NO. *(Include Area Code)* (513) 255-7371	22c. OFFICE SYMBOL WRDC/MTI

EDITION OF 1 JAN 73 IS OBSOLETE

DD FORM 1473, 83 APR

FOREWORD

This technical report covers work performed under Air Force Contract F33600-87-C-0464, DAPro Project. This contract is sponsored by the Manufacturing Technology Directorate, Air Force Systems Command, Wright-Patterson Air Force Base, Ohio. It was administered under the technical direction of Mr. Bruce A. Rasmussen, Branch Chief, Integration Technology Division, Manufacturing Technology Directorate, through Mr. David L. Judson, Project Manager. The Prime Contractor was Integration Technology Services, Software Programs Division, of the Control Data Corporation, Dayton, Ohio, under the direction of Mr. W. A. Osborne. The DAPro Project Manager for Control Data Corporation was Mr. Jimmy P. Maxwell.

The DAPro project was created to continue the development, test, and demonstration of the Integrated Information Support System (IISS). The IISS technology work comprises enhancements to IISS software and the establishment and operation of IISS test bed hardware and communications for developers and users.

The following list names the Control Data Corporation subcontractors and their contributing activities:

SUBCONTRACTOR	ROLE
Control Data Corporation	Responsible for the overall Common Data Model design development and implementation, IISS integration and test, and technology transfer of IISS.
D. Appleton Company	Responsible for providing software information services for the Common Data Model and IDEF1X integration methodology.
ONTEK	Responsible for defining and testing a representative integrated system base in Artificial Intelligence techniques to establish fitness for use.
Simpact Corporation	Responsible for Communication development.

Accession For
NTIS GRA&I ☑
DTIC TAB ☐
Unannounced ☐
Justification
By
Distribution/
Availability Codes
Dist | Avail and/or Special
A-1

Structural Dynamics Research Corporation	Responsible for User Interfaces, Virtual Terminal Interface,and Network Transaction Manager design, development, implementation, and support.
Arizona State University	Responsible for test bed operations and support.

Table of Contents

			Page
SECTION	1.0	INTRODUCTION.....	1-1
SECTION	2.0	PREREQUISITES.....	2-1
SECTION	3.0	DEFINE THE IMPACT PRECOMPILE GROUP.....	3-1
SECTION	4.0	PROCEDURES TO BUILD THE CDM IMPACT EXECUTABLE.....	4-1

SECTION 1

INTRODUCTION

The construction of the CDM Impact Tool requires precompiling of 38 routines for the purpose of discovering the impacts, storing them in the CDM and reporting them on a screen or a printer device. These routines have been grouped into a single logical unit of work. These modules are listed in a test file. After precompilation, the following steps must be executed in order to construct the CDM Impact executable:

- o Generate the CDM Impact Request Processor Main Program
- o Compile and insert into the object library (GENOLB) the generated CDM Impact Request Processor Main Program
- o Create the CDM Impact executable

Section 2 lists the prerequisites of the Impact Environment. Section 3 of this document lists the group to be precompiled. Section 4 contains the step by step instructions of building CDM Impact.

SECTION 2

PREREQUISITES

The Prerequisites to creating the IMPACT ANALYSIS environment are:

1. Existence of an object library IMPAOLB in the directory cdmdir:[tools.Impa] for the Impact software.

2. Existence of an object library GENOLB in the directory cdmdir:[tools.Impa] for the generated code.

3. Existence of a a FORMS directory; this is the directory pointed to by the logical IISSULIB.

4. All the software must be compiled and placed in IMPAOLB according to normal Integration and Testing procedures.

5. The NDDL and NDML executables must be available.

SECTION 3

DEFINE THE IMPACT PRECOMPILE GROUP

This section contains the list of routines to be precompiled as a single logical unit of work. They are contained in the files Impact.tst and Impact2.tst.

Application Name - Impact.tst/IMPACT2.tst

ALGX
ALLDTS
ALLTAGS
AUCX
CATMEM
CIINIT
DBAREA
DBMSDB
DOMAUC
DOMS
ECAUC
ECX
FNDHSDB
GETILUW
GETTAG
HOSTDB
ICHKUN
IGTPKC
IMPACTX
INSAPP
INSCOM
INSOBJ
ISELHP
IVERARA
IVERATT
IVERAUC
IVERDB
IVERDBM
IVERDFD
IVERDI
IVERDOM
IVEROT
IVERENT
IVERHST
IVERKC
IVERLUW
IVEROAC
IVERPSB
IVERRC
IVERRCC
IVERRST
IVERRT
IVERSMD
IVERVEW
KCX
LUWAPP
MODUX

PSBDB
RC
TRACEX
TRAUC
TRDB
TRDF
TRDI
TRRC
TRRT
TRSET
TRUV

SECTION 4

PROCEDURES TO BUILD THE CDM IMPACT EXECUTABLE

The following steps must be executed to construct the CDM Impact executable:

1. Create the Oracle Impact tables in the CDM. Proceed as follows:

   ```
   $ UFI CDM/CDM

   UFI>START ORAIMP.DAT

   UFI>EXIT
   ```

2. Using NDDL, run the Impact meta data into the CDM. Proceed as follows:

   ```
   $ NDDL NDDLIMP.DAT
   ```

   Examine NDDLIMP.OUT to assure that all NDDL commands completed successfully.

3. Precompile and compile the NDML-embedded source code. Proceed as follows:

   ```
   $@BLDIMP
   PRECOMPILE AND COMPILE A GROUP OF PRC'S
   -------------------------------
   NAME OF THE APPLICATION>:IMPACT
   File:  ALGX
   File:  AUCX
   .
   .
   .
   .
   .
   .
   .
   File:  TRUV

   NDML PRECOMPILE SUCCESSFULLY COMPLETED
   BEGIN COMPILING GENERATED CODE
   RESULTS OF COMPILE CAN BE FOUND ON IMPACT.MSG
   ```

4. Repeat step 3 with IMPACT2 as the name of the application.

5. Execute the procedure file LNKIMP.COM to generate the RP-MAIN, compile the RP-MAIN, and place in GENOLB, generate the opt files and link the application.

   ```
   $ @LNKIMP
   ```

5. Run the flan exectuable using the file CDMIMP.FDL. Enter

   [$FLAN CDMIMP.FDL]

   The form definition files will be:

   CDMIMP.FD
   IMPACT1.FD
   IMPACT2.FD
   TRACE1.FD
   TRACE2.FD

6. Define the CDM Impact Function using the user interface utility SYSGEN. The name of the function is CDMIMPACT. The application is GRCDMIMPZZ, and the username, password and role is CDM.

   The following pages contain listings for:

   BLDIMP.COM
   LNKIMP.COM

```
$!
$!     BLDNDDL.COM
$!
$!     RECEIVE A TEST FILE OF PRC'S AND PRECOMPILE AND COMPILE THEM
$!
$WS:= WRITE SYS$OUTPUT
$DEFINE IISSGLIB "CDMDIR:[TOOLS.IMPACT]GENOLB.OLB"
$WS "PRECOMPILE AND COMPILE A GROUP OF PRC'S"
$WS "----------------------------"
$! read an input file containing names of modules to be precompiled
$!
$INQUIRE AP " NAME OF THE APPLICATION>"
$ CREATE 'AP'.DAT
$ OPEN/WRITE   NDMLIN  'AP'.IN
$ OPEN/READ NDDLIN CDMDIR:[TOOLS.IMPACT]'AP'.TST
$NEXT:
$ READ/END_OF_FILE=INDONE NDDLIN FILE
$ WS "FILE:''File'"
$ APPEND 'FILE'.PRC 'AP'.DAT
$ GOTO NEXT
$ INDONE:
$ WRITE NDMLIN "CDMIMP VAX"," ",AP,".DAT ",AP,".ERR ","CDM/CDM"," FD=N"
$ CLOSE NDMLIN
$ CLOSE NDDLIN
$!
$!  INPUTS TO PRECOMPILER ARE NOW SET UP
$!   GO AHEAD AND RUN IT:
$!
$  ASSIGN/USER_MODE SYS$COMMAND SYS$INPUT
$ ASSIGN  'AP'.IN NDML
$ ASSIGN 'AP'.OUT SYS$OUTPUT
$ RUN CMDIR:[RUNAREA]NDML.EXE
$!
$ ALLDONE:
$ DEASSIGN SYS$OUTPUT
$!
$!  check the .out file for errors in precompiling
$!
$OPEN/READ EFLE 'AP'.OUT
$ZR:="0"
$  NERRLOOP:
$   READ/END_OF_FILE=COMPERR EFLE EREC
$     LENG = 'F$LENGTH(EREC)'
$     UN = 'F$LOCATE("UNSUCC",EREC)'
$     IF 'UN' .EQS. 'LENG' THEN GOTO NERRLOOP
$     UN1 = 'UN' - 13
$     UN2 = 'F$EXTRACT(UN1,1,EREC)'
$     IF UN2 .EQS. ZR THEN GOTO NDMLGOOD
$WS "THE PRECOMPILE OF ''AP' HAS ''UN2'  UNSUCCESSFUL ROUTINES"
$WS "CHECK THE ''AP'.ERR FILE FOR ERRORS"
$GOTO EXIT
$COMPERR:
$WS "PRECOMPILE FAILED"
$GOTO EXIT
$!
$!  the precompile was successful, compile the code
$!
$  NDMLGOOD:
```

```
$WS "    "
$WS "NDML PRECOMPILE SUCCESSFULLY COMPLETED"
$WS "BEGIN COMPILING GENERATED CODE"
$  NDMLGDRD:
$ READ/END_OF_FILE = COMPERR EFLE EREC
$  LENG = 'F$LENGTH(EREC)'
$  UN = 'F$LOCATE("COMPILE ALL CODE", EREC)'
$  IF 'UN' .EQS. 'LENG' THEN GOTO NDMLGDRD
$!
$ASSIGN 'AP'.MSG SYS$OUTPUT
$UN1 = 'UN' - 34
$NNAM:='F$EXTRACT(UN1,30,EREC)'
$CLOSE EFLE
$ @'NNAM'
$DEASSIGN SYS$OUTPUT
$WS "RESULTS OF COMPILE CAN BE FOUND ON ''AP'.MSG"
$EXIT:
$DEFINE IISSGLIB "CDMDIR:[TEST]GENOLB.OLB"
```

```
$!
$!        LNKIMP.COM
$!        THIS USES ORACLE VERSION 5.1
$!
$!  CLONED WITH CHANGES 1/20/88 - Richard Stewart
$!  ADD ADDING TO NTM TABLES...NEED TO REMOVE FOR STANDALONE VERSION
$!
$DEFINE/NOLOG TOOLOLB   CDMDIR:[TOOLS.IMPA]IMPAOLB
$DEFINE/NOLOG IISSGLIB "CDMDIR:[TOOLS.IMPA]GENOLB.OLB"
$if p1 .eqs. "N" then goto linkimp
$WS="WRITE SYS$OUTPUT"
$  LINK:
$DEASSIGN SYS$OUTPUT
$WS " "
$WS "Beginning Generation of Rp-Main"
$LUW="CDMIMP"
$CDM="CDM/CDM"
$!
$! generate the rp-main
$! NOTE:  this is done automatically if you link
$!
$!
$!  set up .dat file to send to genrpd
$!
$OPEN/WRITE GENRPD.DAT GENRPD.DAT
$WRITE GENRPD.DAT LUW, " ", CDM
$CLOSE GENRPD.DAT
$ OPEN/WRITE FDLIN FIX.FDL
$ WRITE FDLIN "IDENT   ""23-FEB-1988 09:49:43   VAX-11 FDL Editor"""
$ WRITE FDLIN "    "
$ WRITE FDLIN "SYSTEM"
$ WRITE FDLIN "        SOURCE                  VAX/VMS"
$ WRITE FDLIN "     "
$ WRITE FDLIN "FILE"
$ WRITE FDLIN "        ALLOCATION              3"
$ WRITE FDLIN "        BEST_TRY_CONTIGUOUS     yes"
$ WRITE FDLIN "        EXTENSION               39"
$ WRITE FDLIN "        ORGANIZATION            sequential"
$ WRITE FDLIN "     "
$ WRITE FDLIN "RECORD"
$ WRITE FDLIN "        BLOCK_SPAN              yes"
$ WRITE FDLIN "        CARRIAGE_CONTROL        carriage_return"
$ WRITE FDLIN "        FORMAT                  fixed"
$ WRITE FDLIN "        SIZE                    80"
$ CLOSE FDLIN
$ CONVERT/PAD=%O40/FDL=FIX GENRPD.DAT GENRPD.DAT
$!
$!
$! now run genrpd
$!
$ASSIGN/USER_MODE SYS$COMMAND SYS$INPUT
$ASSIGN 'LUW'.RPD SYS$OUTPUT
$RUNGENRPD
$DEASSIGN SYS$OUTPUT
$ DELETE GENRPD.DAT;*, FIX.FDL;*
$!
$!  now get the needed information to compile the rp-main(s)
$!
$ASSIGN 'LUW'.RDCOMP SYS$OUTPUT
```

```
$GENRPDFLAG = 0
$OPEN/READ  EFLE 'LUW'.RPD
$  RDLOOP:
$READ/END_OF_FILE=ENDMAIN EFLE EREC
$  LENG = 'F$LENGTH(EREC)'
$  DBMS = 'F$LOCATE("FOR DBMS", EREC)'
$  UN   = 'F$LOCATE("STORED ON", EREC)'
$  MN   = 'F$LOCATE("MODULE", EREC)'
$  DB   = 'F$LOCATE("DATA BASE", EREC)'
$  RM   = 'F$LOCATE("REMOTE/", EREC)'
$  HST  = 'F$LOCATE("RUN AT", EREC)'
$ IF 'MN'   .NES. 'LENG' THEN GOTO SAVMODNM
$ IF 'DB'   .NES. 'LENG' THEN GOTO SAVDBN
$ IF 'RM'   .NES. 'LENG' THEN GOTO RMLC
$ IF 'HST'  .NES. 'LENG' THEN GOTO SAVEHST
$ IF 'DBMS' .NES. 'LENG' THEN GOTO SAVEDBMS
$ IF 'UN'   .EQS. 'LENG' THEN GOTO RDLOOP
$GENRPDFLAG = 1
$UN1 = 'UN' + 16
$UNEND = 'F$LOCATE(".",EREC) - UN1
$PL := 'F$EXTRACT(UN1, UNEND, EREC)'
$IF DBMSNM .EQS. "ORACLE" THEN GOTO MAINPCC
$WS "A NEW DBMS TYPE MUST BE ADDED TO THE MAIN COMPILE PART OF THIS PROCE
$GOTO EXIT
$!
$!  get the rp-main mod name
$!
$  SAVMODNM:
$MN1    = 'MN' + 7
$RPMN1 := 'F$EXTRACT(MN1, 10, EREC)'
$LENG   = 'F$LENGTH(RPMN1)'
$MN2    = 'F$LOCATE("ZZZ", RPMN1)
$IF 'MN2' .EQS. 'LENG'  THEN MODLOC = 0
$IF 'MN2' .NES. 'LENG'  THEN MODLOC = 2
$RPMN  := 'F$EXTRACT(MODLOC, 5, RPMN1)
$GOTO RDLOOP
$!
$!  get the remote/local status
$!
$  RMLC:
$RM1 = 'RM' + 13
$RMSW := 'F$EXTRACT(RM1, 1, EREC)'
$GOTO RDLOOP
$!
$!  get the database name
$!
$  SAVDBN:
$DB1 = 'DB' + 10
$DBN := 'F$EXTRACT(DB1, 30, EREC)'
$GOTO RDLOOP
$!
$!  get the host name
$!
$  SAVEHST:
$HST1 = 'HST' + 7
$HSTNM := 'F$EXTRACT(HST1, 3, EREC)'
$GOTO RDLOOP
$!
$!  get the dbms name
```

```
$!
$  SAVEDBMS:
$DBMS1 = 'DBMS' + 9
$DBMSNM := 'F$EXTRACT(DBMS1, 30, EREC)
$GOTO RDLOOP
$!
$!
$!  oracle precompile the rp-main (if needed)
$!
$  MAINPCC:
$PCC INAME='PL'.TMP LNAME='PL'.ERR USERID='CDM' -
 ONAME='PL'.COB INCLUDE=SYS$ORACLE: HOST=COB74 MAXLITERAL=160 REBIND=YES
$ON ERROR THEN WS "ORACLE ERROR IN RP-MAIN ''PL'.TMP"
$ON ERROR THEN GOTO EXIT
$COBOL/ANSI_FORMAT/CHECK=ALL/COPY_LIST/CROSS_REFERENCE/OBJECT='PL'.OBJ -
  /FIPS=74/NOLIST/CHECK=ALL/STANDARD=(SYNTAX)/DEBUG=ALL 'PL'.COB
$ON ERROR THEN WS "COBOL ERROR IN RP-MAIN ''PL'.TMP"
$ON ERROR THEN GOTO EXIT
$LIB/REPLACE IISSGLIB 'PL'.OBJ
$!DELETE 'PL'.OBJ;*
$DELETE 'PL'.COB;*
$DELETE 'PL'.ERR;*
$!
$!  Compile the rpmain.c
$!
$vcc/debug/NOLIST/show=(include)/standard=portable -
      /noopt/OBJECT=CDMDIR:[TOOLS.IMPA]RPMAIN.OBJ/DEFINE=VAX RPMAIN.C
$DELETE RPMAIN.C;*
$GOTO RDLOOP
$!
$!  done generating rp-main
$!
$  ENDMAIN:
$CLOSE EFLE
$DEASSIGN SYS$OUTPUT
$IF GENRPDFLAG .EQ. 0 THEN GOTO MAINERR
$WS " "
$WS  "GENERATION OF REQUEST PROCESSOR MAIN COMPLETE"
$GOTO STARTLINK
$!
$!  there was an error in generating the rp-main
$!
$  MAINERR:
$WS "THE GENRPD HAD ERRORS.  EXAMINE ''LUW'.RPD"
$GOTO EXIT
$STARTLINK:
$WRITE SYS$OUTPUT " - LINKING CDMIMP.EXE "
$ ASSIGN CDMIMP.LINK SYS$OUTPUT
$!
$!
$! inquire p5 "ENTER TWO LETTER NTM DIRECTORY PREFIX "
$! inquire p6 "ENTER NTM CLUSTER FOR THIS RP (T1V OR UIV) "
$!
$P5 = "GR"
$P6 = "UIV"
$P1 = "CDMIMP"
$!
$!  NTMTAB.COM
$!
```

```
$!      15-APR-87
$!      M. DENMAN
$!
$!    UPDATE NTM TABLES APITBL.DAT, APTTBL.DAT, ACTTBL.DAT
$!
$!
$ FLAG=0
$ P7=P5+P1
$ OPEN/READ APITBL.DAT CMDIR:[RUNAREA]APITBL.DAT
$ RD1:
$ READ/END_OF_FILE=CHK APITBL.DAT ENTRY
$ RPND=F$EXTRACT(0,8,ENTRY)
$ IF RPND .NES. P7 THEN GOTO RD1
$ FLAG=1
$ CHK:
$ CLOSE APITBL.DAT
$ IF FLAG .NES. 0 THEN GOTO NOUPD
$ GOTO UPD
$ NOUPD:
$ WRITE SYS$OUTPUT "  "
$ WRITE SYS$OUTPUT "RP MAIN ",P1," ALREADY IN NTM TABLES"
$ WRITE SYS$OUTPUT "  "
$ GOTO LINKIMP
$ UPD:
$ RPAPI=P7+"ZZ"+P6+"1"
$ RPAPT=P1+"ZZ9999010120001130N0"
$ OPEN/APPEND APITBL.DAT CMDIR:[RUNAREA]APITBL.DAT
$ OPEN/APPEND APTTBL.DAT CMDIR:[RUNAREA]APTTBL.DAT
$ WRITE APITBL.DAT RPAPI
$ WRITE APTTBL.DAT RPAPT
$ CLOSE APITBL.DAT
$ CLOSE APTTBL.DAT
$ WRITE SYS$OUTPUT "  "
$ WRITE SYS$OUTPUT "THREE NTM TABLES UPDATED WITH RP ",P1
$ WRITE SYS$OUTPUT "  "
$!
$!DEFINE CDMROLB "CDMDIR:[CDMR]CDMROLB"
$LINKIMP:
$@SYS$ORACLE:LFOR CMDIR:[RUNAREA]CDMIMPZZ -
CDMDIR:[TOOLS.IMPA]CDMIMP.OBJ,-
CDMDIR:[SHARE]SHAROLB/INC=(NDDLYTB),-
CDMDIR:[TOOLS.IMPA]RPMAIN.OBJ,-
CDMDIR:[COM]CDMI/OPTIONS,-
SYS$ORACLE:SQLLIB/LIB,-
CDMDIR:[COM]CDMUI.OPT/OPT,-
CDMDIR:[SHARE]SHAROLB/LIB,-
CDMDIR:[COM]CDMNTM.OPT/OPTIONS SM
$DEASSIGN SYS$OUTPUT
$WRITE SYS$OUTPUT "LINKING COMPLETED"
$EXIT:
$DEFINE/NOLOG IISSGLIB "CDMDIR:[TEST]GENOLB.OLB"
$DEASSIGN TOOLOLB
$!
```


☆U.S. GOVERNMENT PRINTING OFFICE: 1992 - 648-127/62323